

Deadlock

Slide 18

Hệ thống có **12 ổ đĩa từ**, ở thời điểm t_0 yêu cầu về sử dụng ổ đĩa từ như sau:

	Cần tối đa	Đang chiếm giữ	Cần thêm
P_0	10	5	$10-5=5$
P_1	4	2	$4-2=2$
P_2	9	2	$9-2=7$

- Tại t_0 số ổ đĩa còn trống là $12 - (5+2+2) = 3$
- Cấp phát thêm 2 ổ đĩa cho P_1 , P_1 hoàn thành, giải phóng 4 ổ đĩa đang chiếm giữ \rightarrow số ổ đĩa trống : $(3 - 2) + 4 = 5$
- Cấp phát thêm 5 ổ đĩa cho P_0 , P_0 hoàn thành, giải phóng 10 ổ đĩa đang chiếm giữ \rightarrow số ổ đĩa trống : 10
- Cấp phát thêm 7 ổ đĩa cho P_2 , P_2 hoàn thành, giải phóng 9 ổ đĩa đang chiếm giữ

Trạng thái của hệ thống ở thời điểm t_0 là an toàn vì chuỗi $\langle P_1, P_0, P_2 \rangle$ thoả mãn điều kiện an toàn.

Slide 28

5 tiến trình P_0 P_1 P_2 P_3 P_4

3 nguồn tài nguyên:

A (10 đơn vị), B (5 đơn vị), and C (7 đơn vị)

Tại thời điểm t_0 :

	Allocation	Max	Available	Need = Max – Allocation
	A B C	A B C	A B C	A B C
P_0	0 1 0	7 5 3	3 3 2	7 4 3
P_1	2 0 0	3 2 2		1 2 2
P_2	3 0 2	9 0 2		6 0 0
P_3	2 1 1	2 2 2		0 1 1
P_4	0 0 2	4 3 3		4 3 1

Sử dụng thuật toán banker để kiểm tra xem hệ thống ở trạng thái an toàn không ?

1. Khởi tạo

$$Work = Available = [3, 3, 2]$$

$$Finish[i] = FALSE \text{ với } i = 0, 1 \dots 4$$

2. Vì P_1 thỏa mãn $Need[1] \leq Work$ và $Finish[1] = FALSE$ nên cấp phát tài nguyên còn thiếu cho P_1 , P_1 hoàn thành và giải phóng hết tài nguyên đang chiếm giữ:

$$Work = Work + Allocation[1] = [3, 3, 2] + [2, 0, 0] = [5, 3, 2]$$

$$Finish[1] = TRUE$$

(chú ý: ở bước này có thể chọn P_3 thực thi vì P_3 cũng thỏa mãn các điều kiện)

3. Tương tự, lặp lại với P_3 , P_4 , P_2 , P_0 .

Ta có, **<P1, P3, P4, P2, P0>** là chuỗi an toàn → hệ thống đang ở trạng

thái an toàn (*Có thể có nhiều chuỗi an toàn khác*)